Software Testing and Validation – 2017/18

Instituto Superior Técnico

Vos
Project Report

Group 01 – Alameda

Francisca Cambra
ist181057

Rui Ventura
ist181045

April 11, 2018

Contents

1 Method-Scope Tests

1.1 `assignPhoneNumber`

```
final void assignPhoneNumber(int clientNif, int phoneNumber) throws
↪  InvalidOperationException {
```

Assigns a free phone number to a client of *Vos* if all conditions are met. If at least one of these conditions does not hold, then this method does not change anything. In such cases, it throws an `InvalidOperationException` exception.

1.1.1 Test Pattern – Category-partition

1.1.2 Functions

- Primary function
  - Assign free phone number to a client without a number
- Secondary functions
  - Throw `InvalidOperationException` if conditions aren't met
    - Invalid nif (nif $\notin [10^8, 10^9[$)
    - Invalid phone number (number $\notin [10^8, 10^9[$)
    - Client doesn't exist (valid nif)
    - Assign a previously assigned number to a client

1.1.3 Input/Output Parameters

- Input
  - `clientNif` – The nif of the client to assign a number to
  - `phoneNumber` – The phone number to be assigned
  - `clients` – The set of *Vos* clients managed by `ClientManager`
- Output
  - `client` – The updated client, if a number was assigned successfully

1.1.4 Categories & Choices

Parameter	Category	Choices
`clientNif`	*Vos* client (w/ $\#numbers$ phone numbers)	$\#numbers \in [1, 5[$
		$\#numbers = 5$ (MAX)
	Not a *Vos* client	`clientNif` $\in [10^8, 10^9[$
	Invalid nif	`clientNif` $\notin [10^8, 10^9[$
`phoneNumber`	*Vos* phone number	Free (Unassigned)
		Not free (Assigned)
	Not a *Vos* number	`phoneNumber` $\in [10^8, 10^9[$
	Invalid number	`phoneNumber` $\notin [10^8, 10^9[$
`clients`	n-elements	$n = 0$ (Empty)
		$n \in [1, \text{MAX}]$ (Not empty)

1.1.5 Constraints

- Empty `clients` list precludes the possibility of assigning a `phoneNumber`

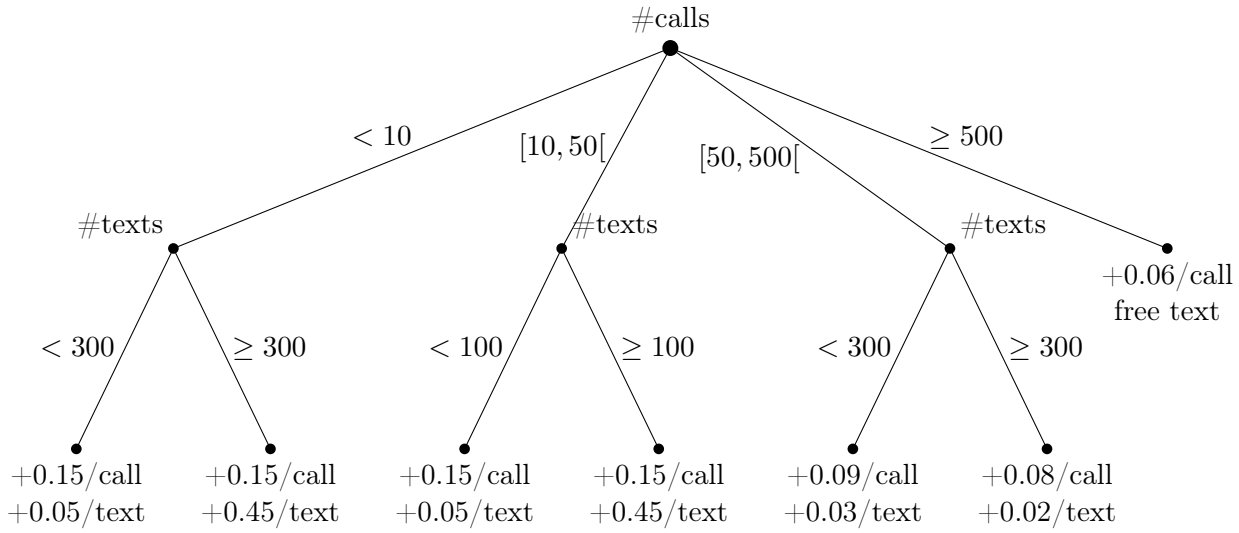
1.1.6 Test Cases

	Choices			Expected Result	
TC	`clientNif`	`phoneNumber`	`clients`	**Exception**	`client`
1	$\#numbers \in [1, 5[$	Free	$n \in [1, \text{MAX}]$	NO	$\#numbers \in]1, 5]$
2	$\#numbers \in [1, 5[$	Not free	$n \in [1, \text{MAX}]$	YES	—
3	$\#numbers \in [1, 5[$	$\notin [10^8, 10^9[$	$n \in [1, \text{MAX}]$	YES	—
4	$\#numbers = 5$	Free	$n \in [1, \text{MAX}]$	YES	—
5	$\#numbers = 5$	Not free	$n \in [1, \text{MAX}]$	YES	—
6	$\#numbers = 5$	$\notin [10^8, 10^9[$	$n \in [1, \text{MAX}]$	YES	—
7	`clientNif` $\in [10^8, 10^9[$	Free	$n \in [1, \text{MAX}]$	YES	—
8	`clientNif` $\in [10^8, 10^9[$	Not free	$n \in [1, \text{MAX}]$	YES	—
9	`clientNif` $\in [10^8, 10^9[$	$\notin [10^8, 10^9[$	$n \in [1, \text{MAX}]$	YES	—
10	`clientNif` $\notin [10^8, 10^9[$	Free	$n \in [1, \text{MAX}]$	YES	—
11	`clientNif` $\notin [10^8, 10^9[$	Not free	$n \in [1, \text{MAX}]$	YES	—
12	`clientNif` $\notin [10^8, 10^9[$	$\notin [10^8, 10^9[$	$n \in [1, \text{MAX}]$	YES	—

1.2 `computeBill` method

The responsibility of `computeBill` method is to determine the value to pay for a client taking into account all communications made by the client through all of his registered mobile phones

1.2.1 Decision Tree



2 Class-Scope Tests

2.1 `Client` class

2.2 `Mobile` class